Amendments to the Claims:

1. (currently amended) A method of normalizing soft decisions obtained from an *M*-ary signal transmitted over a communications channel and subsequently input to an outer decoder, said method comprising the steps of:

generating a first noise power estimate based on a training sequence transmitted along with data over said communications channel;

generating a second noise power estimate derived from the data transmitted over said communications channel;

generating at least one performance based metric based on the reception of said training sequence or on the reception of said data;

calculating a combined noise power estimate as a function of said first noise power estimate, said second noise power estimation and said at least one performance based metric; and

modifying said soft decisions in accordance with said combined noise power estimate so as to yield normalized soft decisions, said normalized soft decisions subsequently input to said outer decoder.

2. (original) The method according to claim 1, wherein said at least one performance metric comprises Signal to Noise Ratio (SNR).

3. (original) The method according to claim 1, wherein said at least one performance metric comprises Bit Error Rate (BER).

4. (original) The method according to claim 1, wherein said first noise power estimate is derived from an ideal training sequence, a received training sequence and a channel estimate.

5. (currently amended) The method according to claim 1, wherein said first noise power estimate is derived from an ideal training sequence, a received training sequence and a channel estimate in accordance with the following expression

$$\hat{\sigma}_{TRN}^2 = \left\| S_{TRN}(k) * \hat{h}(k) - y_{TRN}(k) \right\|$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{TRN}$ is the first noise power estimation;~~

~~$S_{TRN}(k)$ is the ideal training sequence;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~$y_{TRN}(k)$ is the received data sequence.~~

where $\hat{\sigma}^2_{TRN}$ is the first noise power estimation, $S_{TRN}(k)$ is the ideal training sequence, $\hat{h}(k)$ is the channel estimate, $y_{TRN}(k)$ is the received data sequence, k is a discrete time index, the symbol * represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector x and x_k^* is the complex conjugate of x_k.

6. (currently amended) The method according to claim 1, wherein said second noise power estimate is derived from maximum likelihood path metrics generated by a Viterbi Algorithm based inner decoder, a received symbol sequence and a channel estimate in accordance with the following expression

$$\hat{\sigma}^2_{DATA} = \left\|\hat{S}_{DATA}(k) * \hat{h}(k) \quad y(k)\right\|$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{DATA}$ is the second noise power estimation;~~

~~$\hat{S}_{Data}(k)$ is kth symbol from the most likely sequence emerging from the Viterbi Algorithm based inner decoder;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~y(k) is the received training sequence.~~

where $\hat{\sigma}^2_{DATA}$ is the second noise power estimation, $\hat{S}_{Data}(k)$ is k^{th} symbol from the most likely sequence emerging from the Viterbi Algorithm based inner decoder, $\hat{h}(k)$ is the channel estimate, $y(k)$ is the received training sequence, k is a discrete time index, the symbol * represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector x and x_k^* is the complex conjugate of x_k.

7. (currently amended) The method according to claim 1, wherein said second noise power estimate is derived from hard decision error vectors generated by a symbol slicer at the output of an inner decoder, a received symbol sequence and a channel estimate in accordance with the following expression

$$\cancel{\hat{\sigma}^2_{DATA} = \|\hat{S}_{DATA}(k)\hat{h}(k) - y(k)\|}$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{DATA}$ is the second noise power estimation;~~

~~$\hat{S}_{Data}(k)$ is the hard decision symbol most likely to have been transmitted;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~y(k) is the received data sequence.~~

$$\hat{\sigma}^2_{DATA} = \|\hat{S}_{DATA}(k) * \hat{h}(k) - y(k)\|$$

where $\hat{\sigma}^2_{DATA}$ is the second noise power estimation, $\hat{S}_{Data}(k)$ is the hard decision symbol most likely to have been transmitted, $\hat{h}(k)$ is the channel estimate, $y(k)$ is the received data sequence k is a

discrete time index, the symbol * represents the convolution operation, the operator $\|\underline{x}\|$ is defined as the sum of squares of a vector $\underline{x}$ with components x_k as follows

$$\|\underline{x}\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector $\underline{x}$ and x_k^* is the complex conjugate of x_k.

8. (original) The method according to claim 1, wherein said combined noise power estimate is calculated based solely on said first noise estimate in the event said at least one performance metric indicates the variance of said second noise power estimate exceeds that of said first noise power estimate.

9. (original) The method according to claim 1, wherein said combined noise power estimate is calculated based solely on said first noise power estimate in the event said at least one performance metric exceeds a predetermined threshold.

10. (original) The method according to claim 1, wherein said combined noise power estimate is calculated based solely on said second noise power estimate in the event said at least one performance metric indicates the variance of said first noise power estimate exceeds that of said second noise power estimate.

11. (original) The method according to claim 1, wherein said combined noise power estimate is calculated based solely on said second noise power estimate in the event said at least one performance metric exceeds a predetermined threshold.

12. (original) The method according to claim 1, wherein said combined noise power estimate is calculated based on a weighted average of said first noise power estimate and said second noise power estimate.

13. (currently amended) The method according to claim 1, wherein said combined noise power estimate is calculated based on a weighted average of said first noise power estimate and said second noise power estimate in accordance with the following expression

$$\hat{\sigma} = (1-\alpha) * \hat{\sigma}_{TRN} + \alpha * \hat{\sigma}_{Data}$$

where

$\hat{\sigma}$ is the combined noise power estimation; [[and]]

α is the weighting factor, $0 < \alpha < 1$ [[.]] ;

$\hat{\sigma}_{TRN}$ is said first noise power estimation; and

$\hat{\sigma}_{Data}$ is said second noise power estimation.

14. (original) The method according to claim 13, wherein the weighting factor is calculated in accordance with the following

$$0 < (\alpha = k * SNR) < 1/2$$

where k is a constant factor and SNR is the Signal to Noise Ratio.

15. (original) The method according to claim 1, wherein said step of modifying said soft decisions comprises multiplying said soft decisions by said combined noise power estimate to yield normalized soft decisions.

16. (original) The method according to claim 1, wherein said soft decisions are generated by a soft output equalizer.

17. (original) The method according to claim 1, wherein said soft decisions are generated by a combination of hard decision equalizer and soft output generator.

18. (original) The method according to claim 1, wherein said outer decoder comprises a soft input Viterbi Algorithm based convolutional decoder.

19. (currently amended) A method of normalizing soft decisions obtained from an M-ary signal transmitted over a communications channel and subsequently input to an outer decoder, said method comprising the steps of:

generating a first noise power estimate based on a training sequence transmitted along with data over said communications channel;

generating a second noise power estimate derived from the data transmitted over said communications channel;

generating at least one performance based metric based on the reception of said training sequence or on the reception of said data;

calculating a combined noise power estimate solely as a function of said first noise power estimate when said at least one performance metric indicates said first noise power estimate has smaller estimation variance over said second noise power estimate;

calculating said combined noise power estimate solely as a function of said second noise power estimate when said at least one performance metric indicates said second noise power estimate has smaller estimation variance over said first noise power estimate;

modifying said soft decisions in accordance with said combined noise power estimate so as to yield normalized soft decisions, said normalized soft decisions subsequently input to said outer decoder.

20. (original) The method according to claim 19, wherein said at least one performance metric comprises Signal to Noise Ratio (SNR).

21. (original) The method according to claim 19, wherein said at least one performance metric comprises Bit Error Rate (BER).

22. (original) The method according to claim 19, wherein said first noise power estimate is derived from an ideal training sequence, a received training sequence and a channel estimate.

23. (currently amended) The method according to claim 19, wherein said first noise power estimate is derived from an ideal training sequence, a received training sequence and a channel estimate in accordance with the following expression

$$\hat{\sigma}^2_{TRN} = \left\| S_{TRN}(k) * \hat{h}(k) - y_{TRN}(k) \right\|$$

~~where~~ ,

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{TRN}$ is the first noise power estimation;~~

~~$S_{TRN}(k)$ is the ideal training sequence;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~$y_{TRN}(k)$ is the received data sequence.~~

where $\hat{\sigma}^2_{TRN}$ is the first noise power estimation, $S_{TRN}(k)$ is the ideal training sequence, $\hat{h}(k)$ is the channel estimate, $y_{TRN}(k)$ is the received data sequence, k is a discrete time index, the symbol * represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector x and x_k^* is the complex conjugate of x_k.

24. (currently amended) The method according to claim 19, wherein said second noise power estimate is derived from maximum likelihood path metrics generated by a Viterbi Algorithm based inner decoder, a received symbol sequence and a channel estimate in accordance with the following expression

$$\hat{\sigma}^2_{DATA} = \left\|\hat{S}_{DATA}(k) * \hat{h}(k) - y(k)\right\|$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{DATA}$ is the second noise power estimation;~~

~~$\hat{S}_{Data}(k)$ is kth symbol from the most likely sequence emerging from the Viterbi Algorithm based inner decoder;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~y(k) is the received training sequence.~~

where $\hat{\sigma}^2_{DATA}$ is the second noise power estimation, $\hat{S}_{Data}(k)$ is k^{th} symbol from the most likely sequence emerging from the Viterbi Algorithm based inner decoder, $\hat{h}(k)$ is the channel estimate, $y(k)$ is the received training sequence, k is a discrete time index, the symbol * represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|\underline{x}\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector $\underline{x}$ and x_k^* is the complex conjugate of x_k.

25. (currently amended) The method according to claim 19, wherein said second noise power estimate is derived from hard decision error vectors generated by a symbol slicer at the output of an inner decoder, a received symbol sequence and a channel estimate in accordance with the following expression

$$\cancel{\hat{\sigma}^2_{DATA} = \left\|\hat{S}_{DATA}(k)\hat{h}(k) - y(k)\right\|}$$

~~where~~

$$\cancel{\|\underline{x}\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{DATA}$ is the second noise power estimation;~~

~~$\hat{S}_{Data}(k)$ is the hard decision symbol most likely to have been transmitted;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~y(k) is the received data sequence.~~

$$\hat{\sigma}^2_{DATA} = \left\|\hat{S}_{DATA}(k) * \hat{h}(k) - y(k)\right\|$$

where $\hat{\sigma}^2_{DATA}$ is the second noise power estimation, $\hat{S}_{Data}(k)$ is the hard decision symbol most likely to have been transmitted, $\hat{h}(k)$ is the channel estimate, $y(k)$ is the received data sequence k is a discrete time index, the symbol * represents the convolution operation, the operator $\|\underline{x}\|$ is defined as the sum of squares of a vector $\underline{x}$ with components x_k as follows

$$\|\underline{x}\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector $\underline{x}$ and x_k^* is the complex conjugate of x_k.

26. (original) A method of normalizing soft decisions obtained from an *M*-ary signal transmitted over a communications channel and subsequently input to an outer decoder, said method comprising the steps of:

generating a first noise power estimate based on a training sequence transmitted along with data over said communications channel;

generating a second noise power estimate derived from the data transmitted over said channel;

generating at least one performance based metric based on the reception of said training sequence or on the reception of said data;

calculating a combined noise power estimate as a weighted function of said first noise power estimate and said second noise power estimate;

modifying said soft decisions in accordance with said combined noise power estimate so as to yield normalized soft decisions, said normalized soft decisions subsequently input to said outer decoder.

27. (currently amended) The method according to claim 26, wherein said weighted ~~average~~ <u>function</u> is calculated in accordance with the following expression

$$\hat{\sigma} = (1-\alpha) * \hat{\sigma}_{TRN} + \alpha * \hat{\sigma}_{Data}$$

where

$\hat{\sigma}$ is the combined noise power estimation; [[and]]

α is the weighting factor, $0 < \alpha < 1$ [[.]] ;

<u>$\hat{\sigma}_{TRN}$ is said first noise power estimation; and</u>

<u>$\hat{\sigma}_{Data}$ is said second noise power estimation.</u>

28. (original) The method according to claim 27, wherein the weighting factor is calculated in accordance with the following

$$0 < (\alpha = k * SNR) < 1/2$$

where k is a constant factor and SNR is the Signal to Noise Ratio.

29. (original) The method according to claim 26, wherein said at least one performance metric comprises Signal to Noise Ratio (SNR).

30. (original) The method according to claim 26, wherein said at least one performance metric comprises Bit Error Rate (BER).

31. (original) The method according to claim 26, wherein said first noise power estimate is derived from an ideal training sequence, a received training sequence and a channel estimate.

32. (currently amended) The method according to claim 26, wherein said first noise power estimate is derived from an ideal training sequence, a received training sequence and a channel estimate in accordance with the following expression

$$\hat{\sigma}^2_{TRN} = \left\|S_{TRN}(k) * \hat{h}(k) - y_{TRN}(k)\right\|$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_k^K x_k * x_k^*}$$

~~$\hat{\sigma}^2_{TRN}$ is the first noise power estimation;~~

~~$S_{TRN}(k)$ is the ideal training sequence;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~$y_{TRN}(k)$ is the received data sequence.~~

where $\hat{\sigma}^2_{TRN}$ is the first noise power estimation, $S_{TRN}(k)$ is the ideal training sequence, $\hat{h}(k)$ is the channel estimate, $y_{TRN}(k)$ is the received data sequence, k is a discrete time index, the symbol * represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|x\| = \frac{1}{K}\sum_k^K x_k * x_k^*$$

where K is the length of vector x and x_k^* is the complex conjugate of x_k.

33. (currently amended) The method according to claim 26, wherein said second noise power estimate is derived from maximum likelihood path metrics generated by a Viterbi Algorithm based inner decoder, a received symbol sequence and a channel estimate in accordance with the following expression

$$\hat{\sigma}^2_{DATA} = \left\| \hat{S}_{DATA}(k) * \hat{h}(k) - y(k) \right\|$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{DATA}$ is the second noise power estimation;~~

~~$\hat{S}_{Data}(k)$ is kth symbol from the most likely sequence emerging from the Viterbi Algorithm based inner decoder;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~y(k) is the received training sequence.~~

where $\hat{\sigma}^2_{DATA}$ is the second noise power estimation, $\hat{S}_{Data}(k)$ is k^{th} symbol from the most likely sequence emerging from the Viterbi Algorithm based inner decoder, $\hat{h}(k)$ is the channel estimate, $y(k)$ is the received training sequence, k is a discrete time index, the symbol * represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector x and x_k^* is the complex conjugate of x_k.

34. (currently amended) The method according to claim 26, wherein said second noise power estimate is derived from hard decision error vectors generated by a symbol slicer at the output of an inner decoder, a received symbol sequence and a channel estimate in accordance with the following expression

$$\cancel{\hat{\sigma}^2_{DATA} = \left\| \hat{S}_{DATA}(k)\hat{h}(k) - y(k) \right\|}$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{DATA}$ is the second noise power estimation;~~

~~$\hat{S}_{Data}(k)$ is the hard decision symbol most likely to have been transmitted;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~y(k) is the received data sequence.~~

$$\hat{\sigma}^2_{DATA} = \left\| \hat{S}_{DATA}(k) * \hat{h}(k) - y(k) \right\|$$

where $\hat{\sigma}^2_{DATA}$ is the second noise power estimation, $\hat{S}_{Data}(k)$ is the hard decision symbol most likely to have been transmitted, $\hat{h}(k)$ is the channel estimate, $y(k)$ is the received data sequence k is a discrete time index, the symbol * represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector x and x_k^* is the complex conjugate of x_k.

35. (currently amended) A communications receiver for receiving and decoding an M-ary transmitted signal, comprising:

a radio frequency (RF) front end circuit for receiving and converting said ~~M-ary~~ M-ary transmitted signal to a baseband signal;

a demodulator adapted to receive said baseband signal and to generate a received signal therefrom in accordance with the ~~M-ary~~ M-ary modulation scheme used to generate said M-ary transmitted signal;

a first decoder operative to receive said received signal and to generate a sequence of soft symbol decisions therefrom;

a normalization mechanism comprising processing means programmed to:

generate a first noise power estimate based on a training sequence transmitted along with data over ~~said~~ a communications channel;

generate a second noise power estimate derived from the data transmitted over said communications channel;

generate at least one performance based metric based on the reception of said training sequence or on the reception of said data;

calculate a combined noise power estimate as a function of said first noise power estimate, said second noise power estimation and said at least one performance based metric;

modify said soft symbol decisions in accordance with said combined noise power estimate so as to yield normalized soft decisions; and

a second decoder adapted to receive said normalized soft decisions and to generate binary received data therefrom.

36. (original) The receiver according to claim 35, further comprising a speech decoder operative to convert said binary receive data to an audible speech signal.

37. (original) The receiver according to claim 35, further comprising circuit switch data means for converting said binary receive data to a data stream.

38. (original) The receiver according to claim 35, further comprising packet switch data means for converting said binary receive data to a data stream.

39. (currently amended) The receiver according to claim 35, wherein said communications receiver is adapted to receive and decode a Global System for Mobile Communication (GSM) ~~EDGE~~ Enhanced Data Rates for GSM Evolution (EDGE) signal.

40. (currently amended) The receiver according to claim 35, wherein said communications receiver is adapted to receive and decode a ~~GSM-EDGE~~ Global System for Mobile Communication (GSM) Enhanced Data Rates for GSM Evolution (EDGE) Radio Access Network (GERAN) system signal.

41. (original) The receiver according to claim 35, wherein said second decoder comprises a convolutional decoder based on the Viterbi Algorithm (VA).

42. (original) The receiver according to claim 35, wherein said *M*-ary symbol comprises an 8-PSK symbol.

43. (original) The receiver according to claim 35, wherein said first decoder comprises a maximum likelihood sequence estimation (MLSE) equalizer based on the Viterbi Algorithm (VA).

44. (original) The receiver according to claim 35, wherein said first decoder comprises means for performing a sub-optimal complexity reduced maximum likelihood sequence estimation (MLSE) technique based on the Viterbi Algorithm (VA).

45. (original) The receiver according to claim 35, wherein said first decoder comprises a hard decision symbol slicer in combination with a soft output generator.

46. (original) The receiver according to claim 35, wherein said first decoder comprises a Decision Feedback Equalizer (DFE).

47. (original) The receiver according to claim 35, wherein said first decoder comprises a linear type equalizer.

48. (original) An electronic data storage media storing a computer program adapted to program a computer to execute the normalization mechanism process of claim 35.

49. (currently amended) A computer readable storage medium having a computer program embodied thereon for causing a suitably programmed system to normalize soft decisions output by an inner decoder by performing the following steps when such program is executed on said system:

generating a first noise power estimate based on a training sequence transmitted along with data over said communications channel;

generating a second noise power estimate derived from the data transmitted over said <u>communications</u> channel;

generating at least one performance based metric based on the reception of said training sequence or on the reception of said data;

calculating a combined noise power estimate as a function of said first noise power estimate, said second noise power estimation and said at least one <u>performance based</u> metric; and

modifying said soft decisions in accordance with said combined noise power estimate so as to yield normalized soft decisions, said normalized soft decisions subsequently input to said outer decoder.

50. (original) The computer readable storage medium according to claim 49, wherein said at least one performance metric comprises Signal to Noise Ratio (SNR).

51. (original) The computer readable storage medium according to claim 49, wherein said at least one performance metric comprises Bit Error Rate (BER).

52. (original) The computer readable storage medium according to claim 49, wherein said first noise power estimate is derived from an ideal training sequence, a received training sequence and a channel estimate.

53. (currently amended) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to derive said first noise power estimate from an ideal training sequence, a received training sequence and a channel estimate in accordance with the following expression

$$\hat{\sigma}_{TRN}^2 = \left\| S_{TRN}(k) * \hat{h}(k) - y_{TRN}(k) \right\|$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}_{TRN}^2$ is the first noise power estimation;~~

~~$S_{TRN}(k)$ is the ideal training sequence;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~$y_{TRN}(k)$ is the received data sequence.~~

where $\hat{\sigma}_{TRN}^2$ is the first noise power estimation, $S_{TRN}(k)$ is the ideal training sequence, $\hat{h}(k)$ is the channel estimate, $y_{TRN}(k)$ is the received data sequence, k is a discrete time index, the symbol * represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector x and x_k^* is the complex conjugate of x_k.

54. (currently amended) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to derive said second noise power estimate from

maximum likelihood path metrics generated by a Viterbi Algorithm based inner decoder, a received symbol sequence and a channel estimate in accordance with the following expression

$$\hat{\sigma}^2_{DATA} = \left\| \hat{S}_{DATA}(k) * \hat{h}(k) - y(k) \right\|$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{DATA}$ is the second noise power estimation;~~

~~$\hat{S}_{Data}(k)$ is kth symbol from the most likely sequence emerging from the Viterbi Algorithm based inner decoder;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~y(k) is the received training sequence.~~

where $\hat{\sigma}^2_{DATA}$ is the second noise power estimation, $\hat{S}_{Data}(k)$ is k^{th} symbol from the most likely sequence emerging from the Viterbi Algorithm based inner decoder, $\hat{h}(k)$ is the channel estimate, $y(k)$ is the received training sequence, k is a discrete time index, the symbol $*$ represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector x and x_k^* is the complex conjugate of x_k.

55. (currently amended) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to derive said second noise power estimate from hard decision error vectors generated by a symbol slicer at the output of an inner decoder, a received symbol sequence and a channel estimate in accordance with the following expression

$$\cancel{\hat{\sigma}^2_{DATA} = \left\| \hat{S}_{DATA}(k)\hat{h}(k) - y(k) \right\|}$$

~~where~~

$$\cancel{\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*}$$

~~$\hat{\sigma}^2_{DATA}$ is the second noise power estimation;~~

~~$\hat{S}_{Data}(k)$ is the hard decision symbol most likely to have been transmitted;~~

~~$\hat{h}(k)$ is the channel estimate; and~~

~~y(k) is the received data sequence.~~

$$\hat{\sigma}^2_{DATA} = \left\|\hat{S}_{DATA}(k) * \hat{h}(k) - y(k)\right\|$$

where $\hat{\sigma}^2_{DATA}$ is the second noise power estimation, $\hat{S}_{Data}(k)$ is the hard decision symbol most likely to have been transmitted, $\hat{h}(k)$ is the channel estimate, $y(k)$ is the received data sequence k is a discrete time index, the symbol * represents the convolution operation, the operator $\|x\|$ is defined as the sum of squares of a vector x with components x_k as follows

$$\|x\| = \frac{1}{K}\sum_{k}^{K} x_k * x_k^*$$

where K is the length of vector x and x_k^* is the complex conjugate of x_k.

56. (original) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to calculate said combined noise power estimate based solely on said first noise estimate in the event said at least one performance metric indicates the variance of said second noise power estimate exceeds that of said first noise power estimate.

57. (original) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to calculate said combined noise power estimate based solely on said first noise power estimate in the event said at least one performance metric exceeds a predetermined threshold.

58. (original) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to calculate said combined noise power estimate based solely on said second noise power estimate in the event said at least one performance metric indicates the variance of said first noise power estimate exceeds that of said second noise power estimate.

59. (original) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to calculate said combined noise power estimate based solely on said second noise power estimate in the event said at least one performance metric exceeds a predetermined threshold.

60. (original) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to calculate said combined noise power estimate based on a weighted average of said first noise power estimate and said second noise power estimate.

61. (currently amended) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to calculate said combined noise power estimate based on a weighted average of said first noise power estimate and said second noise power estimate in accordance with the following expression

$$\hat{\sigma} = (1-\alpha) * \hat{\sigma}_{TRN} + \alpha * \hat{\sigma}_{Data}$$

where

$\hat{\sigma}$ is the combined noise power estimation; [[and]]

α is the weighting factor, $0 < \alpha < 1$ [[.]] ;

$\hat{\sigma}_{TRN}$ is said first noise power estimation; and

$\hat{\sigma}_{Data}$ is said second noise power estimation.

62. (original) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to calculate the weighting factor in accordance with the following

(original) $0 < (\alpha = k * SNR) < 1/2$

where k is a constant factor and SNR is the Signal to Noise Ratio.

63. (original) The computer readable storage medium according to claim 49, wherein said computer program is suitably programmed to modify said soft decisions by multiplying said soft decisions by said combined noise power estimate to yield normalized soft decisions.